

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

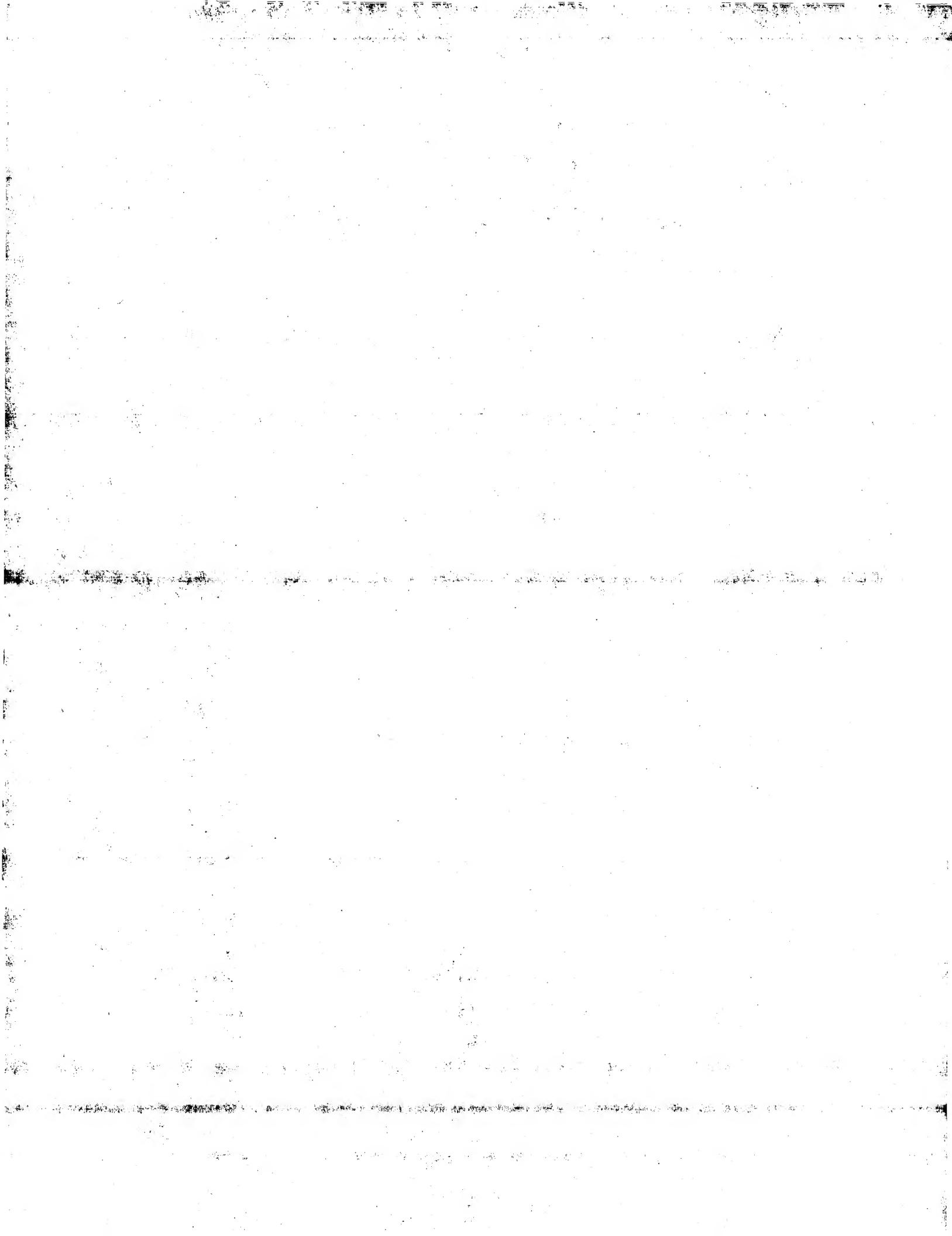
Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.**





(19)日本国特許庁 (JP)

## (12) 公開特許公報 (A)

(11)特許出願公開号  
特開2001-282505  
(P2001-282505A)

(43)公開日 平成13年10月12日 (2001.10.12)

(51)Int.Cl'	識別記号	F I	マーコード*(参考)
G 0 6 F 7/00		F 0 2 D 45/00	3 7-6 B 3 G 0 8 4
F 0 2 D 45/00	3 7 6	G 0 6 F 11/00	3 5 0 P. 5 B 0 2 2
G 0 6 F 11/00	3 5 0	7/00	1 0 1 Z

審査請求 未請求 請求項の数7 O L (全10頁)

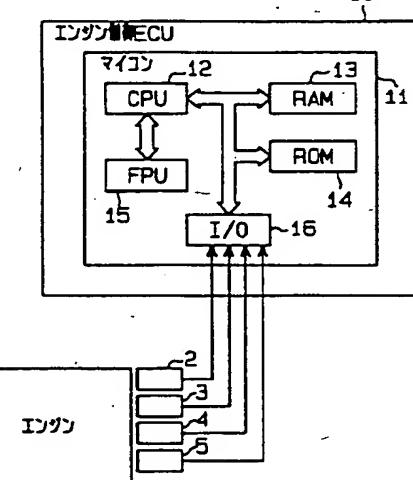
(21)出願番号	特願2000-101028(P2000-101028)	(71)出願人	000004260 株式会社デンソー 愛知県刈谷市昭和町1丁目1番地
(22)出願日	平成12年4月3日 (2000.4.3)	(72)発明者	杉村 卓俊 愛知県刈谷市昭和町1丁目1番地 株式会社デンソー内
		(74)代理人	100068755 弁理士 山田 博宣 (外1名)
		F ターム(参考)	3G084 DA04 DA26 EA01 EB10 5B022 BA01 BA03 CA03 DA06 DA09 EA06 FA11

## (54)【発明の名稱】 浮動小数点演算機能を有した電子制御装置

## (57)【要約】

【課題】浮動小数点演算機能を有した電子制御装置において、非数の発生に伴う制御不良を未然に防止する。

【解決手段】マイクロコンピュータ11は、浮動小数点演算を実施するための浮動小数点演算プロセッサ(FPU)15と、CPU12等を有し、所定の制御プログラムに従い各種制御を実施する。CPU12は、浮動小数点演算に用いられる浮動小数点型データが非数である旨を判定したとき、同浮動小数点演算に代わるバックアップ処理を実施する。CPU12が実施するバックアップ処理としては、FPU15を用いることなく、浮動小数点型データに代わる整数型データを代入したり、或いは整数型データを用いた演算を実施したりする。



【特許請求の範囲】

【請求項 1】 浮動小数点型データを演算するための浮動小数点演算機能を有し、所定の制御プログラムに従い各種制御を実施する電子制御装置において、

浮動小数点型データが非数であるか否かを判定する非数判定手段と、

前記非数判定手段により非数である旨が判定されたとき、浮動小数点演算に代わるバックアップ処理を実施するバックアップ手段と、を備えることを特徴とする電子制御装置。

【請求項 2】 前記非数判定手段は、浮動小数点演算毎にその際に用いられる浮動小数点型データが非数であるか否かを判定することを特徴とする請求項 1 に記載の電子制御装置。

【請求項 3】 前記非数判定手段は、浮動小数点演算毎に影響を与える浮動小数点型データが非数であるか否かを判定することを特徴とする請求項 1 に記載の電子制御装置。

【請求項 4】 前記非数判定手段は、全ての浮動小数点型データのうちいずれかが 1 つでも非数であるか否かを判定することを特徴とする請求項 1 に記載の電子制御装置。

【請求項 5】 浮動小数点型データが非数であれば、その旨を表す非数判定フラグをその都度セットし、前記非数判定手段は、非数判定フラグを参照して非数か否かを判定し、前記バックアップ手段は、前記非数判定フラグの判定結果に応じてバックアップ処理を実施することを特徴とする請求項 1～4 のいずれか一項に記載の電子制御装置。

【請求項 6】 前記バックアップ手段は、浮動小数点型データに代えて整数型データをバックアップ値として用いるようにしたことを特徴とする請求項 1～5 のいずれか一項に記載の電子制御装置。

【請求項 7】 前記バックアップ手段は、浮動小数点演算に代えて整数型データを用いた演算を実施することを特徴とする請求項 1～5 のいずれか一項に記載の電子制御装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、浮動小数点演算機能を有した電子制御装置に関するものである。

【0002】

【従来の技術】 一般に、自動車のエンジン制御などに適用される電子制御装置（ECU）では、従来より整数型データ（固定小数点型データ）を用いて各種演算を実施していたが、近年では浮動小数点演算プロセッサ（FPU: Floating-Point Unit）の導入により、浮動小数点型データによる演算が実施できるようになった。浮動小数点型データによれば、固定小数点型データに比べて非常に細かい精度で演算結果が得られる。

【0003】 浮動小数点型データは例えば IEEE 754 規格に従い構成され、図 10 (a) に示されるようにその内訳として、1 ビットの符号部と、8 ビットの指数部と、23 ビットの仮数部とを有する。こうして仮数部が 23 ビットで構成される 4 バイト（単精度記憶形式）の浮動小数点型データの場合、7 衔の分解能（0.0000001 の分解能）を持つ。

【0004】 また、図 10 (b) には、単精度記憶形式でのビットパターンが示され、浮動小数点型データは、指数部と仮数部の組み合わせにより正規化数、非正規化数、無限大、ゼロ及び非数に区別される。ここで、非数以外は数値を表し、非数は数値でないことを表す。例えば、0/0 や +∞-∞ といった数値として表現できない演算結果を表す場合に非数が用いられる。

【0005】

【発明が解決しようとする課題】 上記の通り浮動小数点型データには非数というデータ形式があり、非数を含む、例えば四則演算の結果は全て非数となり無効となる。例えば、比較演算において非数が 1 以上か以下かを比較した場合、その結果は何れも偽となる。そのため、電子制御装置内で非数が発生した場合、その演算結果（出力値）は全く保障できないという問題がある。

【0006】 エンジン制御において非数が発生する条件としては主に 2 つ考えられる。一つ目の条件は、電子制御装置の動作中やバッテリバックアップ中にノイズにより浮動小数点型の RAM 値が書き換わって RAM 値そのものが非数に変わるものである。例えば、ノイズにより浮動小数 RAM 値が F F F F F F F F h (全ビット 1) に書き換わる場合等である。二つ目の条件は、浮動小数点演算に用いられる引数がノイズ等によって書き換わり、0/0 のような演算を行って二次的に非数が発生する場合である。

【0007】 ここで、図 11 に示すエンジン回転数の演算処理にてその事例を説明する。つまり、図 11 の処理では、エンジンの 360° CA (クランク角) の回転に要した時間 T 360 を算出してそれを FR 0 値として記憶し（ステップ 900）、FR 0 = 0 でないことを条件に、「1 sec」を FR 1 に、「60」を FR 2 にセットする（ステップ 910～930）。

【0008】 その後、演算式

$$FR 0 = FR 1 / FR 0 \times FR 2$$

により、FR 0 値を算出し（ステップ 940）、FR 0 値をエンジン回転数 Ne とする（ステップ 950）。

【0009】 上記図 11 の処理において、仮にステップ 930 の直後にノイズ等により FR 0, FR 1 値が各々 0 に変化した場合を想定する。この場合、ステップ 940 では「0/0」の演算が行われ、ステップ 940 の演算結果である FR 0 値が非数となる。その結果、エンジン回転数が正しく算出できることとなる。

【0010】 また例えば、燃料噴射量の演算式は、次

式のように示される。

$$f = F_{base} \times f_{HL}$$

ここで、 $F_{base}$ は、整数型データとして算出された基本噴射量であり、 $f_{HL}$ は、浮動小数点型データとして算出された負荷補正量である。この演算において、負荷補正量 $f_{HL}$ が非数である場合、算出結果の燃料噴射量 $f$ の値も非数となり、正常な燃料噴射が実施できなくなる。

【0011】本発明は、上記問題に着目してなされたものであって、その目的とするところは、浮動小数点演算機能を有した電子制御装置において、非数の発生に伴う制御不良を未然に防止することができる電子制御装置を提供することである。

【0012】

【課題を解決するための手段】請求項1に記載の発明によれば、非数判定手段により、浮動小数点型データが非数であるか否かが判定され、非数である旨が判定されたとき、バックアップ手段により、浮動小数点演算に代わるバックアップ処理が実施される。つまり、浮動小数点演算において、非数を含むデータで実施される演算結果は全て非数となるため、正しい制御データを得ることができなくなるが、本発明では、浮動小数点型データが非数である場合、浮動小数点演算に代わるバックアップ処理が実施される。これにより、非数の発生に伴う制御不良を未然に防止することができる。

【0013】

請求項2に記載の発明によれば、浮動小数点演算毎にその際に用いられる浮動小数点型データが非数であるか否かが判定されるので、その時々の浮動小数点演算が誤りなく実施される。つまり、非数を含む演算により無効となる浮動小数点演算を確実に判定できる。

【0014】

請求項3に記載の発明によれば、浮動小数点演算毎に影響を与える浮動小数点型データが非数であるか否かが判定される。この場合、浮動小数点演算に影響を与える浮動小数点型データとは、例えば浮動小数点演算の前提条件として用いられるデータであり、この浮動小数点型データが非数の時に、本来の浮動小数点演算とは異なるバックアップ処理を行うことにより、誤って浮動小数点演算が実施され制御性が悪化するといった不都合を回避できる。

【0015】

請求項4に記載の発明によれば、全ての浮動小数点型データのうちいずれか1つでも非数であるか否かが判定される。つまり、浮動小数点型データのうちのいずれか1つでも非数となる際には、その発生要因（具体的には、ノイズ）がある環境下でマイクロコンピュータが動作していると判断できる。従って、その際には、いつ非数が発生してもおかしくないので、全ての浮動小数点演算が禁止されるとともに、その浮動小数点演算に代わるバックアップ処理が実施される。

【0016】

請求項5に記載の発明によれば、浮動小数点演算にて得られた浮動小数点型データが非数であれ

ば、その旨を表す非数判定フラグがセットされ、非数判定手段によって、非数判定フラグが参照されることで非数か否かが判定される。そして、非数判定フラグの判定結果によりバックアップ処理が実施される。この場合、同じ浮動小数点型データを複数の浮動小数点演算に用いる場合や、請求項4に記載のように全ての浮動小数点型データのうちいずれか1つでも非数であるか否かを判定する場合に適用すれば、処理を簡素化できるので実用上好ましいものとなる。

【0017】また、バックアップ手段が実行するバックアップ処理としては、請求項6に記載のように、浮動小数点型データに代えて整数型データをバックアップ値として用いる。但し、バックアップ値としては、制御に支障のない値を用いる。この場合、本来、浮動小数点演算により得られる制御データに比べるとその精度は劣るが、非数発生時に制御が中断されることなく継続できる。

【0018】さらに、請求項7に記載のように、浮動小数点演算に代わる整数型データを用いた演算を実施するようとしてもよい。この場合、浮動小数点型データを用いた浮動小数点演算と比較して演算結果の精度は劣るが、整数型データに応じた制御データをバックアップ値として算出できるので、制御性の悪化を抑制できる。

【0019】

【発明の実施の形態】以下、この発明を具体化した実施の形態を図面に従って説明する。図1は、エンジン制御システムの概略を示すブロック図である。本システムにおいて、車載エンジン1は、例えばガソリン噴射式多気筒内燃機関として構成される。

【0020】エンジン制御ECU10はマイクロコンピュータ（以下、マイコンという）11を備え、そのマイコン11には中央演算処理装置（CPU）12と、読み書き可能な記憶装置（RAM）13と、読み取り専用記憶装置（ROM）14と、浮動小数点演算プロセッサ（FPU）15と、入出力装置（I/O）16とが設けられている。ここで、FPU15は浮動小数点形式の演算を実施し、CPU12は浮動小数点形式以外の演算を実施する。また、I/O16には周知のA/D変換器が含まれる。

【0021】ECU10には、エンジン1に設けられたセンサ群からエンジン運転状態を表す各種情報が入力される。センサ群は、例えばエンジン回転数を検出するための回転数センサ2、吸気管圧力を検出するための吸気圧センサ3、冷却水温を検出するための水温センサ4、排ガス中の酸素濃度から空燃比（A/F）を検出するためのA/Fセンサ5などとなる。そして、ECU10は、前記入力されるセンサ信号を基に、図示しないインジクタによる燃料噴射の制御や、点火装置による点火時期の制御などを実施する。

【0022】FPU15により演算される浮動小数点型

データは、既述の通り例えば IEEE 754 規格に従い構成され、半精度記憶形式であれば図 10 (a) に示すデータ形式を有する。また、図 10 (b) の如く、浮動小数点型データは、指数部と仮数部との組み合わせにより正規化数、非正規化数、無限大、ゼロ及び非数に区別される。

【0023】本実施の形態では、FPU 15 による浮動小数点演算の結果（浮動小数点型データ）が非数になった場合に、その旨を判定し、浮動小数点演算を実施しないようにしている。具体的に、FPU 15 を用いた浮動小数点演算を実施する処理としては、燃料噴射量の算出処理や、点火時期の算出処理があり、例えば、燃料噴射量  $f$  は次式のように算出される。

$$[0024] f = F_{base} \times f_{HL}$$

ここで、 $F_{base}$  は、整数型データとして算出された基本噴射量であり、 $f_{HL}$  は浮動小数点型データとして算出された負荷補正量である。従って、FPU 15 を用いて燃料噴射量  $f$  を算出する際に、負荷補正量  $f_{HL}$  が非数である場合、その演算結果（噴射量  $f$  の値）も非数となってしまう。このため、本実施の形態では、負荷補正量  $f_{HL}$  が非数である場合には、FPU 15 を用いた燃料噴射量  $f$  の演算を禁止し、それに代わるバックアップ処理にて燃料噴射量  $f$  を求めるようにしている。

【0025】ここで、CPU 12 により実行される処理のうち、負荷補正量  $f_{HL}$  の算出処理を図 2 を用い、燃料噴射量算出処理を図 3 を用いて説明する。なお、図 2 の処理は、所定時間毎に実施される時間同期処理であり、図 3 の処理は、回転数センサ 2 のパルス信号に基づき、所定クランク角毎に実施されるクランク同期処理である。

【0026】図 2 に示すように、CPU 12 は、ステップ 100 にてエンジン回転数が 3000 rpm 以上であるか否かを判定し、3000 rpm 以上であれば、ステップ 110 に移行する。ステップ 110 にて CPU 12 は、エンジン水温が 60°C 以上であるか否かを判定し、60°C 以上であれば、ステップ 120 に移行する。そして、同ステップ 120 にて、FPU 15 を用い次式のように負荷補正量  $f_{HL}$  を求める。

$$[0027] f_{HL} = f_{Ne} \times MAF \times f_{Gain}$$

ここで、 $f_{Ne}$  は、回転数センサ 2 のパルス信号に基づき浮動小数点型データとして算出されたエンジン回転数であり、MAF は、吸気圧センサ 3 の検出値であって、吸気管圧力（吸気圧）に基づく整数型データの電圧値である。また、 $f_{Gain}$  は、浮動小数点型データで設定された定数である。

【0028】このように、負荷補正の前提条件として、回転数が 3000 rpm 以上、かつ水温が 60°C 以上の条件が成立した場合に FPU 15 による浮動小数点演算が実施され、浮動小数点型データの負荷補正量  $f_{HL}$  が算出される。

【0029】一方、回転数が 3000 rpm 未満、或いは水温が 60°C 未満の場合には、ステップ 100 或いはステップ 110 にて否定判定され、ステップ 130 において CPU 12 は、負荷補正量  $f_{HL}$  に固定値として浮動小数点型データである「1.0」を代入した後本処理を終了する。

【0030】次に、噴射量算出処理を図 3 を用いて説明する。先ず、CPU 12 は、ステップ 200 にて、エンジン回転数と吸気圧とにに基づき整数型データの基本噴射量  $F_{base}$  を算出し、続くステップ 210 にて、負荷補正量  $f_{HL}$  が非数であるか否かを判定する。つまり、既述のように、負荷補正量  $f_{HL}$  は、浮動小数点型データとして求められており、その補正量  $f_{HL}$  の bit 3 0 ~ 2 3 が「1 1 1 1 1 1 1 1」で、かつ、補正量  $f_{HL}$  の bit 2 2 ~ 0 のうちいずれかが「1」であるか否かを判定する。そして、ステップ 210 にて否定判定された場合、CPU 12 はステップ 220 に移行して、FPU 15 を用い、次式の浮動小数点演算にて燃料噴射量  $f$  を算出する。

$$[0031] f = F_{base} \times f_{HL}$$

つまり、燃料噴射量  $f$  は、基本噴射量  $F_{base}$  に負荷補正量  $f_{HL}$  を乗算することにより算出され、その値は浮動小数点型データとして算出される。その後、ステップ 230 にて CPU 12 は、算出値が非数か否かを判定し、算出値が非数でない旨を判定した場合、浮動小数点演算にて得られた燃料噴射量  $f$  の値が正しいものとして本処理を終了する。

【0032】一方、ステップ 210 またはステップ 230 にて肯定判定された場合、CPU 12 は、ステップ 240 に移行して、FPU 15 を用いることなく整数型データの基本噴射量  $F_{base}$  を燃料噴射量  $f$  に代入した後本処理を終了する。

【0033】次いで、点火時期  $e$  の算出処理を図 4 を用いて説明する。なお、同処理にて使用される負荷補正量  $e_{HL}$  も前記燃料噴射量  $f$  の算出処理における負荷補正量  $f_{HL}$  と同様に浮動小数点演算により求められた浮動小数点型データである。

【0034】先ず、CPU 12 は、ステップ 300 にて、エンジン回転数と吸気圧とにに基づき整数型データの基本点火時期  $E_{base}$  を算出し、続くステップ 310 にて、負荷補正量  $e_{HL}$  が非数であるか否かを判定する。そして、ステップ 310 にて否定判定された場合、CPU 12 はステップ 320 に移行して、FPU 15 を用い次式の浮動小数点演算にて点火時期  $e$  を算出する。

$$[0035] e = E_{base} \times e_{HL}$$

つまり、点火時期  $e$  は、基本点火時期  $E_{base}$  に負荷補正量  $e_{HL}$  を乗算することにより算出され、その値は浮動小数点型データとして算出される。その後、ステップ 330 にて CPU 12 は、算出値が非数か否かを判定し、算出値が非数でない旨を判定した場合、浮動小数点演算にて得られた点火時

期 $e$ の値が正しいものとして本処理を終了する。

【0036】一方、ステップ310またはステップ330にて肯定判別された場合、CPU12は、ステップ340に移行して、FPU15を用いることなく整数型データの基本点火時期 $E_{base}$ を点火時期 $e$ に代入した後本処理を終了する。

【0037】なお、本実施の形態では、図3のステップ210, 230、図4のステップ310, 330の処理が非数判定手段に相当し、図3のステップ240、図4のステップ340の処理がバックアップ手段に相当する。

【0038】以上詳述した本実施の形態によれば、以下に示す効果が得られる。

(1) 浮動小数点演算に使用される浮動小数点型データ(負荷補正量 $f_{HL}$ ,  $e_{HL}$ )が非数である場合、その浮動小数点演算に代わるバックアップ処理が実施される。この場合、浮動小数点演算において非数を含む四則演算の結果は非数となり無効となるが、浮動小数点演算に用いられる浮動小数点型データが非数であるとき、その無効となる浮動小数点演算を禁止することができる。このようにすれば、制御データとしての燃料噴射量 $f$ 、点火時期 $e$ が非数となることを回避でき、非数の発生に伴う制御不良を未然に防止することができる。

【0039】(2) 本実施の形態では、非数の発生時のバックアップ処理として、整数型データの基本噴射量 $F_{base}$ を燃料噴射量 $f$ に、整数型データの基本点火時期 $E_{base}$ を点火時期 $e$ に代入するようにした。つまり、制御データのバックアップ値として制御に支障のない値を代入するようにしている。このようにすれば、バックアップ処理に伴うROM容量の増加を抑制できる。

【0040】以下に、本発明における第2～第4の実施の形態を説明する。但し、第2～第4の実施の形態におけるECU10の構成は、図1に示す第1の実施の形態と同じであり、CPU12により実行される処理が第1の実施の形態と異なる。

【0041】(第2の実施の形態) 本発明における第2の実施の形態を図5及び図6を用いて説明する。上記第1の実施の形態では、浮動小数点演算に使用される浮動小数点型データ(負荷補正量 $f_{HL}$ )が非数であるか否かを判定するものであったが、本実施の形態では、演算に使用される浮動小数点型データではなく、その演算に影響を与える浮動小数点型データが非数であるか否かを判定している。また、本実施の形態において、負荷補正量 $f_{HL}$ は、加速時、減速時で異なる演算式で算出されるようになっており、加速または減速の判定は、負荷変化量に相当するパラメータ $f_{DL}$ により判定される。同パラメータ $f_{DL}$ は、エンジン回転数及び吸気圧に基づいて算出された浮動小数点型データであって、パラメータ $f_{DL}$ が正の値であれば加速を意味し、負の値であれば減速を意味する。

【0042】ここで、本実施の形態における負荷補正量 $f_{HL}$ の算出処理を図5を用いて説明する。図5のフローチャートは、前記図2の一部を変更したものであり、図5のステップ100, 110, 130は共通の処理を示す。

【0043】要するに、エンジン回転数が3000 rpm未満、または水温が60°C未満のいずれかの条件が成立した場合(ステップ100、ステップ110において否定判別された場合)、ステップ130にて負荷補正量 $f_{HL}$ に固定値として浮動小数点型データである「1.0」を代入し本処理を終了する。

【0044】一方、エンジン回転数が3000 rpm以上、かつ水温が60°C以上である場合、ステップ400にて、CPU12は、パラメータ $f_{DL}$ が非数であるか否かを判定する。そして、パラメータ $f_{DL}$ が非数でない旨が判定された場合、CPU12はステップ410に移行して、パラメータ $f_{DL}$ を用いて加速または減速かを判定する。ここで、パラメータ $f_{DL}$ が「0」よりも大きい値であれば、ステップ420に進みFPU15を用い加速時の浮動小数点演算( $f_{Ne} \times MAF \times f_{Gain1}$ )を実施して、浮動小数点型データの負荷補正量 $f_{HL}$ を算出した後本処理を終了する。また、パラメータ $f_{DL}$ が「0」以下の値であれば、ステップ430に進みFPU15を用い減速時の浮動小数点演算( $f_{Ne} \times MAF \times f_{Gain2}$ )を実施して、浮動小数点型データの負荷補正量 $f_{HL}$ を算出した後本処理を終了する。なお、 $f_{Gain1}$ は、浮動小数点型データで設定された加速用の定数であり、 $f_{Gain2}$ は、浮動小数点型データで設定された減速用の定数である。

【0045】一方、ステップ400にてパラメータ $f_{DL}$ が非数である旨が判定された場合は、ステップ440に移行して、整数型データの固定値「1」を負荷補正量 $f_{HL}$ に代入した後本処理を終了する。なおここで、整数型データの負荷補正量 $f_{HL}$ は、RAM13において、浮動小数点型データの負荷補正量 $f_{HL}$ とは別に確保された記憶領域に格納される。

【0046】次に、本実施の形態における噴射量算出処理を図6を用いて説明する。図6のフローチャートは、前記図3の一部を変更したものであり、図6のステップ200, 220, 230, 240は共通の処理を示す。

【0047】つまり、ステップ200にて基本噴射量 $F_{base}$ を算出した後、ステップ500にて、負荷変化量に相当するパラメータ $f_{DL}$ が非数であるか否かを判定する。そして、パラメータ $f_{DL}$ が非数である旨が判定された場合、CPU12はステップ510にて、FPU15を用いることなく整数型データの固定値「1」を代入した負荷補正量 $F_{HL}$ と基本噴射量 $F_{base}$ とを乗算することにより燃料噴射量 $f$ (=  $F_{base} \times F_{HL}$ )を算出する。

【0048】一方、ステップ500にてパラメータ $f_{DL}$

Lが非数でない旨が判定された場合、CPU12はステップ220にて、FP15を用いた浮動小数点演算にて燃料噴射量 $f$  (=F<sub>base</sub>×f<sub>HL</sub>)を算出する。そして、CPU12はステップ230にて、算出値が非数であるか否かを判定し、非数でなければ本処理を終了し、非数であれば、ステップ240にて、燃料噴射量 $f$ として基本噴射量F<sub>base</sub>を代入した後本処理を終了する。

【0049】なお、本実施の形態では、図5のステップ400、図6のステップ500及びステップ230の処理が非数判定手段に相当し、図5のステップ440、図6のステップ510及びステップ240の処理がバックアップ手段に相当する。

【0050】要するに、浮動小数点演算に影響を与える浮動小数点データ、即ち、浮動小数点演算の前提条件として用いられる浮動小数点型データが非数である場合、前提条件を正しく判定できずに誤って浮動小数点演算が実施されるおそれがある。これに対し、本実施の形態では、前提条件として用いられる浮動小数点型データが非数である旨が判定されたとき、浮動小数点演算に代わる処理が実施される。その結果、誤って浮動小数点演算が実施され不的確な制御データが算出されることでエンジン1の制御性が悪化するといった不都合を回避できる。

【0051】(第3の実施の形態)次に、本発明における第3の実施の形態を図7及び図8を用いて説明する。本実施の形態では、全ての浮動小数点型データのうちいずれか一つでも非数となったとき、全ての浮動小数点演算が禁止され、それに代わるバックアップ処理が実施されるようになっている。

【0052】その具体例として、先ず、燃料噴射量 $f$ の算出処理を図7を用いて説明する。図7のフローチャートは、前記図3の一部を変更したものであり、図7のステップ200, 220, 230, 240は共通の処理を示す。

【0053】図7では、非数判定フラグXALLを用いており、同フラグXALLは、全ての浮動小数点型データのうちいずれかが一つでも非数となったときに「1」がセットされる。例えば、負荷補正量 $f_{HL}$ や $e_{HL}$ 等を算出するための浮動小数点演算の処理(図示略)において、浮動小数点演算の算出値が非数である場合、非数判定フラグXALLに「1」がセットされる。

【0054】図7のフローチャートについて詳述すると、ステップ200にて基本噴射量F<sub>base</sub>を算出した後、CPU12はステップ600に移行して、非数判定フラグXALLが「1」であるか否かを判定する。そして、非数判定フラグXALLが「0」である旨が判定されたとき、CPU12はステップ220に移行して、FP15を用い浮動小数点演算にて燃料噴射量 $f$  (=F<sub>base</sub>×f<sub>HL</sub>)を算出するとともに、ステップ230にて算出値が非数であるか否かを判定する。そして、ステップ230にて、演算値が非数でない旨を判定したとき

本処理を終了し、非数である旨を判定したとき、ステップ610に移行して非数判定フラグXALLに「1」をセットする。そして、ステップ240に移行し燃料噴射量 $f$ として基本噴射量F<sub>base</sub>を代入した後本処理を終了する。

【0055】また、非数判定フラグXALLに「1」がセットされた場合、ステップ600にて肯定判定され、ステップ240にて燃料噴射量 $f$ として基本噴射量F<sub>base</sub>を代入した後本処理を終了する。つまり、非数判定フラグXALLに「1」がセットされると、ステップ220の浮動小数点演算を巡回して、浮動小数点演算に代わるバックアップ処理として、ステップ240の処理が実施される。

【0056】次に、点火時期 $e$ の算出処理を図8を用いて説明する。図8のフローチャートは、前記図4の一部を変更したものであり、図8のステップ300, 320, 330, 340は共通の処理を示す。

【0057】図8の処理においても、ステップ320の浮動小数点演算を行う前に、ステップ700にて、非数判定フラグXALLが「1」であるか否かを判定する。そして、非数判定フラグXALLが「1」である旨を判定したとき、ステップ320の浮動小数点演算を巡回し、ステップ320の浮動小数点演算に代わるバックアップ処理として、ステップ340にて点火時期 $e$ に基本点火時期E<sub>base</sub>をセットする。また、ステップ320の浮動小数点演算による算出値が非数である旨を判定したとき、ステップ710に移行して非数判定フラグXALLに「1」をセットした後、ステップ340の処理を実施する。

【0058】つまり、本実施の形態では、エンジン制御に用いられる全ての浮動小数点演算において、その演算結果が非数である旨が判定されたとき、共通の非数判定フラグXALLをセットするようしている。そして、浮動小数点演算について、その演算が実施される前に、非数判定フラグXALLがセットされているか否かを判定し、そのフラグXALLがセットされていれば、全ての浮動小数点演算を禁止して、それに代わるバックアップ処理を実施するようしている。

【0059】なお、本実施の形態では、図7のステップ230、ステップ600、図8のステップ330、ステップ700の処理が非数判定手段に相当し、図7のステップ240、図8のステップ340の処理がバックアップ手段に相当する。

【0060】実際に、浮動小数点型データのうちいずれかが一つでも非数となる際には、その発生要因(具体的には、ノイズ)がある環境下でマイコン11が動作していると判断できる。従って、その際には、いつ非数が発生してもおかしくないので、全ての浮動小数点演算が禁止されるとともに、その浮動小数点演算に代わるバックアップ処理が実施される。

【0061】また、本実施の形態では、非数判定フラグ X ALL を用いて、非数であるか否かを判定するようにしているので、全ての浮動小数点データについてその値により非数であるか否かを判定する場合と比較して、処理を簡素化でき実用上好ましいものとなる。

【0062】上述の通り、いずれかの浮動小数点データで非数が発生し、それに伴いフラグ X ALL がセットされた場合において、ノイズ等の非数発生要因が解消され、浮動小数点データが正常な値に戻った時には、フラグ X ALL をリセットして本来の浮動小数点演算を再開するよう構成してもよい。これにより、ノイズ等に起因して一時的に非数が発生する際に、バックアップ処理が長時間継続されることではなく、高精度な浮動小数点演算に復帰できる。

【0063】(第4の実施の形態) 次に、本発明における第4の実施の形態を図9を用いて説明する。図9のフローチャートは、前記図2の一部を変更したものであり、図10のステップ100～130は共通の処理を示す。

【0064】詳述すると、エンジン回転数が 3000 rpm 未満、水温が 60°C 未満のいずれかの条件が成立した場合(ステップ100、ステップ110において否定判定された場合)、ステップ130にて負荷補正量  $\pm HL$  に固定値として浮動小数点型データである「1.0」を代入し本処理を終了する。一方、エンジン回転数が 3000 rpm 以上、かつ水温が 60°C 以上である場合(ステップ100、ステップ110のいずれも肯定判定された場合)、CPU12 はステップ800に進み、浮動小数点型データとして算出されたエンジン回転数  $\pm Ne$  が非数であるか否かを判定する。そして、エンジン回転数  $\pm Ne$  が非数でない旨を判定したとき、ステップ120にて、FP15 を用い浮動小数点演算を実施して、浮動小数点型データの負荷補正量  $\pm HL$  ( $\pm Ne \times MAF \times \pm Gain$ ) を算出した後に本処理を終了する。一方、エンジン回転数  $\pm Ne$  が非数である旨を判定したとき、CPU12 はステップ810にて、FP15 を用いることなく、整数型データとして算出されたエンジン回転数  $Ne$  と整数型データの吸気圧  $MAF$  をパラメータとしたマップデータの補間演算を実施する。そして、CPU12 は、整数型データとして得られた補間値を変換することにより浮動小数点型データとして負荷補正量  $\pm HL$  に代入し、本処理を終了する。

【0065】なお、本実施の形態では、ステップ800の処理が非数判定手段に相当し、ステップ810の処理がバックアップ手段に相当する。このように、本実施の形態では、浮動小数点型データが非数である旨が判定されたとき、浮動小数点演算に代えて、整数型データを用いた演算が実施される。この場合、浮動小数点演算と比較して演算結果の精度は劣るが、整数型データに対応した制御データをバックアップ値として算出できるので、

実用上好ましいものとなる。

【0066】なお本発明は、上記以外に次の形態にて具体化できる。上記第1の実施の形態において、図3のステップ210では、浮動小数点型データの負荷補正量  $\pm HL$  の値により非数であるか否かを判定していたが、フラグを用いて判定してもよい。つまり、図2のステップ120における浮動小数点演算の算出値が非数である場合、負荷補正量  $\pm HL$  が非数である旨を示す非数判定フラグをセットし、図3のステップ210にて、同フラグを参照することによって負荷補正量  $\pm HL$  が非数か否かを判定する。同様に、図4のステップ310、図5のステップ400等の処理においても、非数判定フラグを用い浮動小数点型データが非数であるか否かを判定してもよい。この場合、非数判定フラグを参照することによって、浮動小数点型データが非数であるか否かを判定できる。よって、同じ浮動小数点型データを複数の浮動小数点演算に用いる場合に適用すれば、処理を簡素化できるので実用上好ましいものとなる。

【0067】上記第2の実施の形態では、浮動小数点演算に影響を与える浮動小数点型データとして、パラメータ  $\pm DL$  に具体化したがこれに限定するものではない。例えば、浮動小数点演算を実施するための前提条件として、空燃比補正量等の浮動小数点型データを用いる場合、その空燃比補正量が非数であるか否かを判定し、非数であれば浮動小数点演算を禁止して、それに代わるバックアップ処理を実施する。

【0068】また例えば、浮動小数点演算毎に影響を与える浮動小数点型データが非数であるか否かを判定するものとして、非数発生時に、点火制御、噴射制御等の各種制御毎に区分して非数判定を行うようとする。具体的には、点火制御、噴射制御等の各種制御毎に、共通の非数判定フラグを用意し、例えば、点火制御の浮動小数点演算にて、非数が発生したとき点火制御用の非数判定フラグをセットし、その非数判定フラグを用いて、点火制御にて実施される全ての浮動小数点演算を禁止する。このようにしても、非数の発生により浮動小数点演算に悪影響が及ぶことを防止できる。

【0069】上記各実施の形態では、単精度記憶形式の浮動小数点型データを扱う電子制御装置(ECU)10について例示したが、倍精度記憶形式の浮動小数点型データを扱う電子制御装置にも適用できる。

【0070】以上の説明では自動車用エンジン制御を例に本発明を述べたが、このように自動車の走行に関与する電子制御装置に本発明を適用することで、自動車の制御システムの信頼性を高めることができる。

【図面の簡単な説明】

【図1】発明の実施の形態におけるエンジン制御システムの概要を示すブロック図。

【図2】第1の実施の形態における負荷補正量算出処理を示すフローチャート。

【図 3】第1の実施の形態における噴射量算出処理を示すフローチャート。

【図 4】第1の実施の形態における点火時期算出処理を示すフローチャート。

【図 5】第2の実施の形態における負荷補正量算出処理を示すフローチャート。

【図 6】第2の実施の形態における噴射量算出処理を示すフローチャート。

【図 7】第3の実施の形態における噴射量算出処理を示すフローチャート。

【図 8】第3の実施の形態における点火時期算出処理を示すフローチャート。

【図 9】第4の実施の形態における負荷補正量算出処理を示すフローチャート。

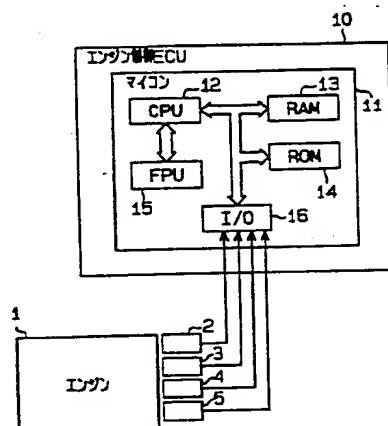
【図 10】浮動小数点型データの構成を示す図。

【図 11】回転数演算処理を示すフローチャート。

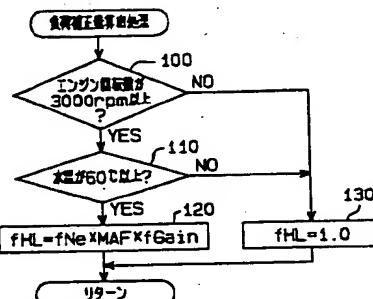
【符号の説明】

10…エンジン制御ECU（電子制御装置）、11…マイコン、12…CPU、13…RAM、14…ROM、15…FPU、16…I/O

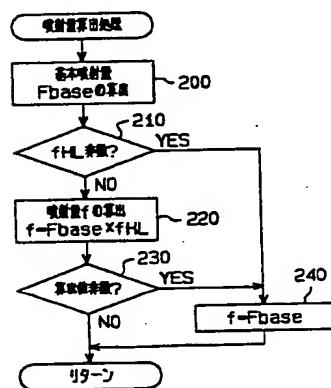
【図 1】



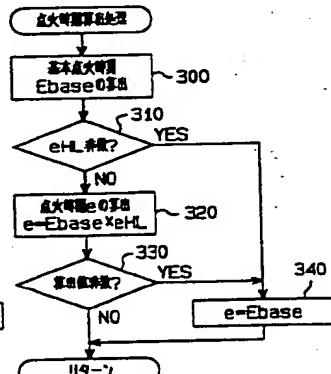
【図 2】



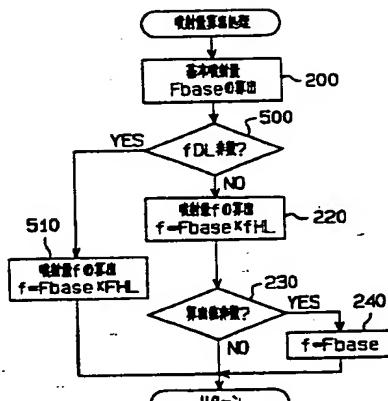
【図 3】



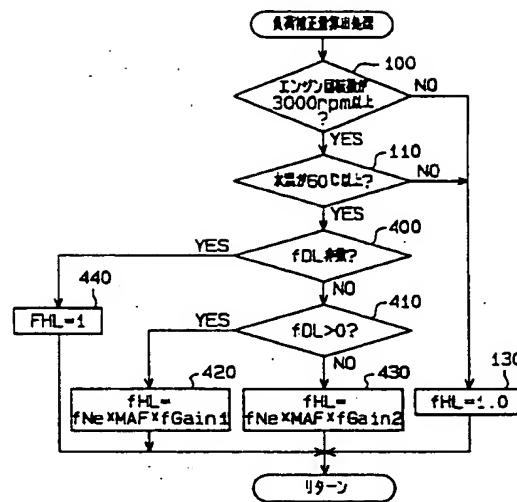
【図 4】



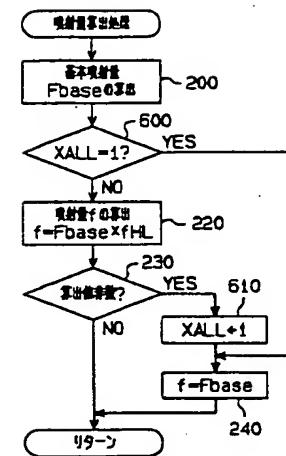
【図 6】



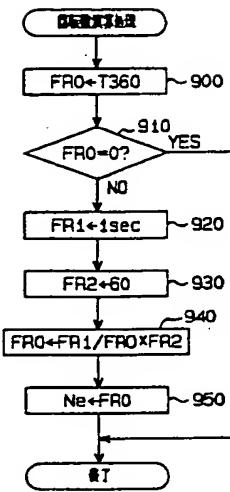
【図 5】



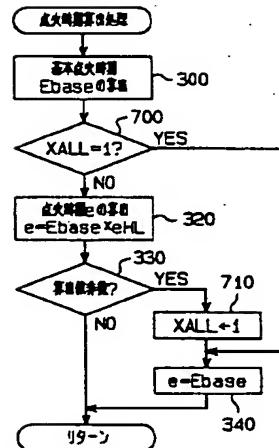
【図 7】



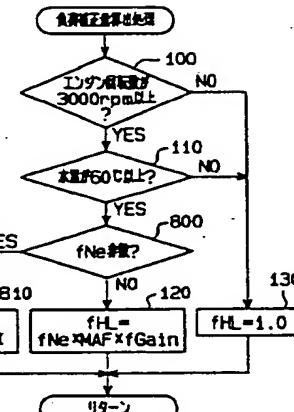
【図 11】



【図 8】

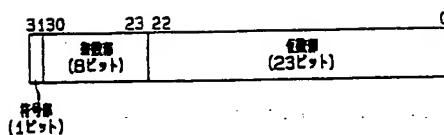


【図 9】



【図 10】

(a)



(b)

符号数	値数部	値数の意味
255	0以外	余数
	0	正または負の整数大
254-1	-	正整数
0	0以外	余正整数
	0	正または負のゼロ